# Analysis of successful putts from various distances of professional golfers

We are analysis successful putts from various distances of professional golfers. The data is given to us in a form of distance, no of tries for that distance, and number of successes for that distance.

It was chosen to compare 4 generalised linear models where for both we have considered the response variable to be the number of successes, which naturally fits binomial distribution:

Full Code can be found in [Appendix](#_Appendix) where JAGS code for Models can be found at [golfers\_bin1.model](#_golfers_bin1.model), [golfers\_bin2.model](#_golfers_bin2.model). R code is at [R Code](#_R_Code).

Both models were created using three chains and samples were drawn using 10000 iterations.

For each of the models, we have generated DIC values and Table 1 contains the results.

|  |  |
| --- | --- |
| Model | DIC |
| Model 1 | 2520 |
| Model 2 | 365.9 |
| Model 3 | 185.8 |
| Model 4 | 155.4 |

Table 1. DIC Values

In this case the clear choice is Model 4 due to lowest DIC value.

Mean values for each of the parameters are:

The next step was to run cross-correlation. We can see on Figure 1 that there is a strong negative correlation between and , and between and . One approach to fix this is to potentially do reparameterization which is not done for this analysis due to the low DIC value.

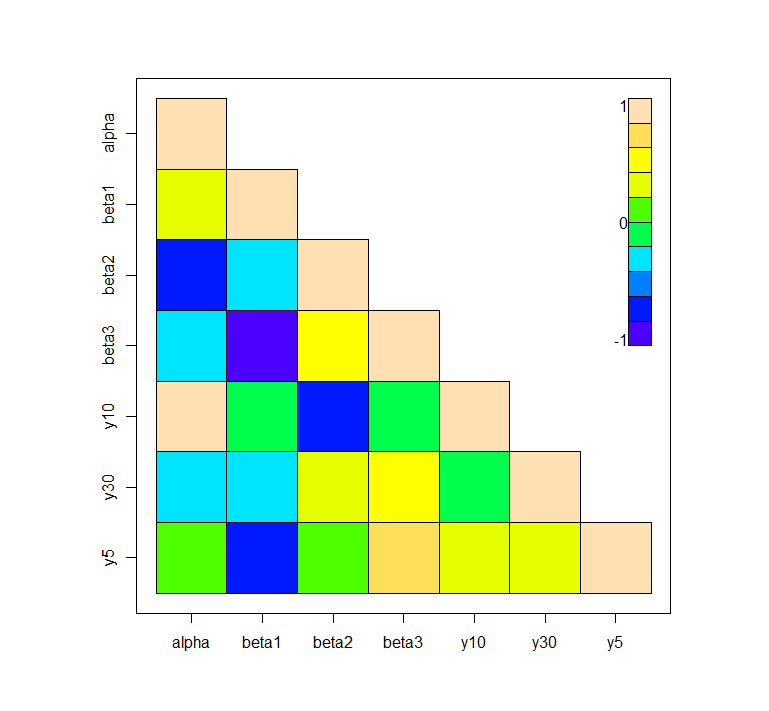


Figure 1. Cross Correlation Plot

Finally, we have completed estimates of proportion of success for 5, 10 and 30 feet distance attempts and the results are in the Table 2 including 95% credible interval.

|  |  |  |
| --- | --- | --- |
| Distance | Estimate | 95% credible interval |
| 5 | 0.65491896 | (6.329279e-01, 0.676514972) |
| 10 | 0.31418449 | (2.931234e-01, 0.335689129) |
| 30 | 0.00070438 | (2.436157e-06, 0.004912875) |

The estimates for 5 and 10 feet are slightly higher than the numbers we have from the input data (0.589 and 0.335) and are out of credible interval which brings us to believe that we could parameterize this model to improve it.

# Appendix

## R Code

require(rjags)

# the data

golfers\_data=read.delim("golfers.dat")

golfers\_data$ProportionSuccess = golfers\_data$Successes / golfers\_data$Tries

golfers.data = list(

N=nrow(golfers\_data),

Distance=golfers\_data$Distance,

Tries=golfers\_data$Tries,

Successes = golfers\_data$Successes)

# Models and sampling

golfers.model\_bin1=jags.model("golfers\_bin1.model", golfers.data, n.chains=3)

golfers.samps\_bin1=coda.samples(golfers.model\_bin1,variable.names=c("alpha", "y5", "y10", "y30"),1e4)

golfers.model\_bin2=jags.model("golfers\_bin2.model", golfers.data, n.chains=3)

golfers.samps\_bin2=coda.samples(golfers.model\_bin2,variable.names=c("alpha","beta", "y5", "y10", "y30"),1e4)

golfers.model\_bin3=jags.model("golfers\_bin3.model", golfers.data, n.chains=3)

golfers.samps\_bin3=coda.samples(golfers.model\_bin3,

variable.names=c("alpha","beta1", "beta2", "y5", "y10", "y30"),1e4)

golfers.model\_bin4=jags.model("golfers\_bin4.model", golfers.data, n.chains=3)

golfers.samps\_bin4=coda.samples(golfers.model\_bin4,

variable.names=c("alpha","beta1", "beta2", "beta3",

"y5", "y10", "y30"),1e4)

# DIC

golfers.dic\_bin1 = dic.samples(golfers.model\_bin1, 1e4)

golfers.dic\_bin2 = dic.samples(golfers.model\_bin2, 1e4)

golfers.dic\_bin3 = dic.samples(golfers.model\_bin3, 1e4)

golfers.dic\_bin4 = dic.samples(golfers.model\_bin4, 1e4)

golfers.dic\_bin1

golfers.dic\_bin2

golfers.dic\_bin3

golfers.dic\_bin4

# Sumamry statistics

golfers.summary\_bin1 <- summary(golfers.samps\_bin1)

golfers.means\_bin1 <- golfers.summary\_bin1$statistics[,"Mean"]

golfers.summary\_bin2 <- summary(golfers.samps\_bin2)

golfers.means\_bin2 <- golfers.summary\_bin2$statistics[,"Mean"]

golfers.summary\_bin3 <- summary(golfers.samps\_bin3)

golfers.means\_bin3 <- golfers.summary\_bin3$statistics[,"Mean"]

golfers.summary\_bin4 <- summary(golfers.samps\_bin4)

golfers.means\_bin4 <- golfers.summary\_bin4$statistics[,"Mean"]

# Print means

golfers.means\_bin1

golfers.means\_bin2

golfers.means\_bin3

golfers.means\_bin4

# Print quantiles

golfers.summary\_bin1$quantiles

golfers.summary\_bin2$quantiles

golfers.summary\_bin3$quantiles

golfers.summary\_bin4$quantiles

par(mfrow=c(1,1))

crosscorr(golfers.samps\_bin4)

crosscorr.plot(golfers.samps\_bin4)

## golfers\_bin1.model

model{

for (i in 1:N){

Successes[i] ~ dbin(p[i], Tries[i])

logit(p[i]) <- alpha

}

alpha ~ dnorm(0, 0.0001)

p\_temp5 <- alpha

p\_temp10 <- alpha

p\_temp30 <- alpha

y5 <- 1 / (1 + exp(-p\_temp5))

y10 <- 1 / (1 + exp(-p\_temp10))

y30 <- 1 / (1 + exp(-p\_temp30))

}

## golfers\_bin2.model

model{

for (i in 1:N){

Successes[i] ~ dbin(p[i], Tries[i])

logit(p[i]) <- alpha + beta\*(Distance[i]-mean(Distance[]))

}

alpha ~ dnorm(0, 0.0001)

beta ~ dnorm(0, 0.0001)

p\_temp5 <- alpha + beta\*(5-mean(Distance[]))

p\_temp10 <- alpha + beta\*(10-mean(Distance[]))

p\_temp30 <- alpha + beta\*(30-mean(Distance[]))

y5 <- 1 / (1 + exp(-p\_temp5))

y10 <- 1 / (1 + exp(-p\_temp10))

y30 <- 1 / (1 + exp(-p\_temp30))

}

## golfers\_bin3.model

model{

for (i in 1:N){

Successes[i] ~ dbin(p[i], Tries[i])

logit(p[i]) <- alpha + beta1\*(Distance[i]-mean(Distance[])) + beta2\*pow(Distance[i]-mean(Distance[]),2)

}

alpha ~ dnorm(0, 0.0001)

beta1 ~ dnorm(0, 0.0001)

beta2 ~ dnorm(0, 0.0001)

p\_temp5 <- alpha + beta1\*(5-mean(Distance[])) + beta2\*pow(5-mean(Distance[]),2)

p\_temp10 <- alpha + beta1\*(10-mean(Distance[])) + beta2\*pow(10-mean(Distance[]),2)

p\_temp30 <- alpha + beta1\*(30-mean(Distance[])) + beta2\*pow(30-mean(Distance[]),2)

y5 <- 1 / (1 + exp(-p\_temp5))

y10 <- 1 / (1 + exp(-p\_temp10))

y30 <- 1 / (1 + exp(-p\_temp30))

}

## golfers\_bin4.model

model{

for (i in 1:N){

Successes[i] ~ dbin(p[i], Tries[i])

logit(p[i]) <- alpha + beta1\*(Distance[i]-mean(Distance[])) + beta2\*pow(Distance[i]-mean(Distance[]),2) + beta3\*pow(Distance[i]-mean(Distance[]),3)

}

alpha ~ dnorm(0, 0.0001)

beta1 ~ dnorm(0, 0.0001)

beta2 ~ dnorm(0, 0.0001)

beta3 ~ dnorm(0, 0.0001)

p\_temp5 <- alpha + beta1\*(5-mean(Distance[])) + beta2\*pow(5-mean(Distance[]),2) + beta3\*pow(5-mean(Distance[]),3)

p\_temp10 <- alpha + beta1\*(10-mean(Distance[])) + beta2\*pow(10-mean(Distance[]),2) + beta3\*pow(10-mean(Distance[]),3)

p\_temp30 <- alpha + beta1\*(30-mean(Distance[])) + beta2\*pow(30-mean(Distance[]),2) + beta3\*pow(30-mean(Distance[]),3)

y5 <- 1 / (1 + exp(-p\_temp5))

y10 <- 1 / (1 + exp(-p\_temp10))

y30 <- 1 / (1 + exp(-p\_temp30))

}